CSCI 5352
Final Project Proposal: Predictive Toxicology Framework

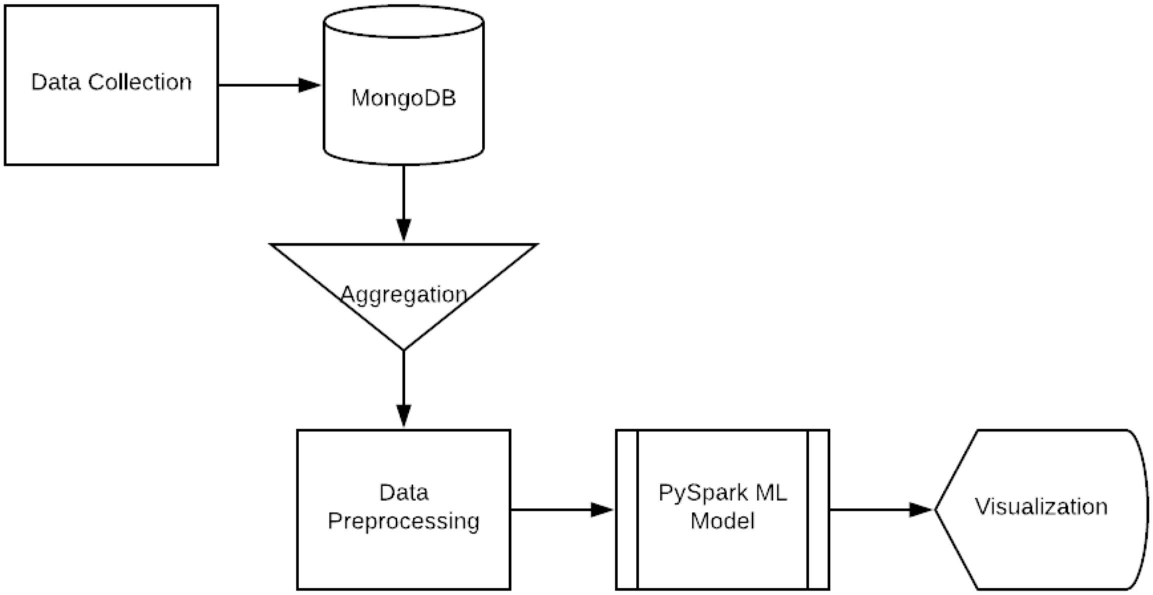
Students

- Ignacio Tripodi
- Harshini Priya Muthukrishnan
- Chu-Sheng Ku
- Chi Chen

Problem Description

The area of computational predictive toxicology has been increasingly expanding over the last few years due to recent toxicity assay requirements by regulatory agencies, especially in the European Union. The large volume of chemicals requiring a toxicity assessment and the possible different chemical endpoints (skin irritation, eye irritation, systemic toxicity, etc) doesn't scale to perform traditional assays, particularly animal models. This, in combination with a push towards replacing animal models due to humane reasons, makes computational predictions increasingly more useful as a pre-screening tool. Regulatory agencies are in a constant search for different computational models to parse the vast amounts of data available, and come up with classifiers to evaluate which untested chemicals are at a higher risk of posing a threat of eliciting a toxic response (or, which have a high likelihood of being completely safe at normal exposure levels).

Architecture



- Data collection: Download the dataset from NTP and CTD and store them to S3
- MongoDB: Use Python code to extract the data and save them to the database
- Aggregation: Use Spark Mongo connector to get the multiple tables from the database. Refer to Dataset portion of the proposal
- Data Preprocessing: Collect only data required for the classifier and clean it
- Machine learning model: Build a machine learning model on Spark. Refer to description below for details.
- Visualization: Visualize the analysis result to explain the useful knowledge found

Final Project Layout

The project will consist of two main components: data aggregation using cloud computing tools we've learned during the course of this class, and a machine learning framework to attempt prediction of untested chemicals.

Machine Learning

Once we have these merged data (and the rest of chemicals in CTD that are not included in the NTP datasets, to use as testing) we will train a classifier to determine whether the rest of CTD chemicals will pose a high or low risk of toxicity, for each of the three endpoints (dermal, oral, and inhalation), based on the genes and pathways they affect, how they affect them, and potentially additional properties of the chemical itself. We will employ the SciKit Learn library for Python and test various classifiers to determine which display the best performance. The performance evaluation will be based on cross-validation using a random fraction of NTP's chemical list as testing, since we don't have a "ground truth" for the rest of chemicals. Our predictions can be shared with the scientific community for further evaluation.

Dataset

We will employ the full dataset of acute toxicity (dermal, oral, and inhalation) from the National Toxicology Program (NTP) which contains a list of compounds, their active chemical components, and various other metadata. The toxicity classification score will be used as the "label" to be predicted by our supervised learning setting. We will decide over the course of the project whether we will make this a multi-label classification problem, or if we define a threshold to determine whether a compound is "worth testing" or not (e.g., positive if it scores higher than 4 in OECD's scale which ranges from 1 to 6). The latter could also allow us to perform a linear regression instead. For each of these chemical compounds and their individual components, we will attempt to merge some information from the Comparative Toxicogenomics Database (CTD). Specifically, the chemical-gene and chemical-pathway relations. We can use the information extracted from CTD as features (which genes the chemical is known to interact with, and which biochemical pathway steps is known to participate in). This will be equivalent to a large "right join", where we will only consider those compounds or small molecules that have an entry in CTD. As a stretch goal, we could also obtain "fingerprints" from each of these chemicals, to also include as features. These are boolean or real-valued vectors that are generated from specialized libraries like RDkit, that reflect properties of the chemical (molecular weight, hydrophobicity, number of aromatic rings, etc). This first step could be performed using Hadoop or Spark.

- What format is it in - Structured csv
- Does the data need significant preprocessing? Yes, the dataset come from different sources and not compatible to each other. We have to aggregate, preprocess and clean them before we start to analyze them.
- Describe dataset: static

- How will the data be accessed? We could download the dataset from NTP and CTD website.
- Where will the data be stored and how? We will download the dataset and store them on S3 and collect the information to MongoDB by python scripting process.

Challenges

- The problem is hard to solve because we do not have many experts both master in computer science and biology.
- Most of us are lack of biological domain knowledge for finding the relations among all the dataset.
- We have to figure out which features are useful to build the machine learning model, and feature engineering is always a challenge for mining something useful.
- The chemical data may have a class imbalance, as the distribution of OECD toxicity codes may not be uniform.
- Data preprocessing and cleaning could be time-consuming since we need to integrate different datasets, which have incompatible formats.
- Integration of the different components in the architecture could also be a problem considering the huge size of the dataset.
- Unsure of how we want to show the results at the end. Hence visualization could be a challenge.

Timeline

- Data collection - 11/1
- Data aggregation architecture (stretch goal: incorporate RDkit features) - 11/8
- Machine learning architecture - 11/15
- Classifier model - 11/15
- Hyperparameter optimization & reclassification - 11/20
- Toxicity prediction of the rest of CTD chemicals - 11/22
- Visualization - 11/29

Responsibilities

- Ignacio Tripodi - Data Aggregation, ML Architecture, Optimization
- Chu-Sheng Ku - Classifier Model & ML Architecture
- Chi Chen - Classifier Model & Optimization
- Harshini Muthukrishnan - Toxicity Prediction & Visualization